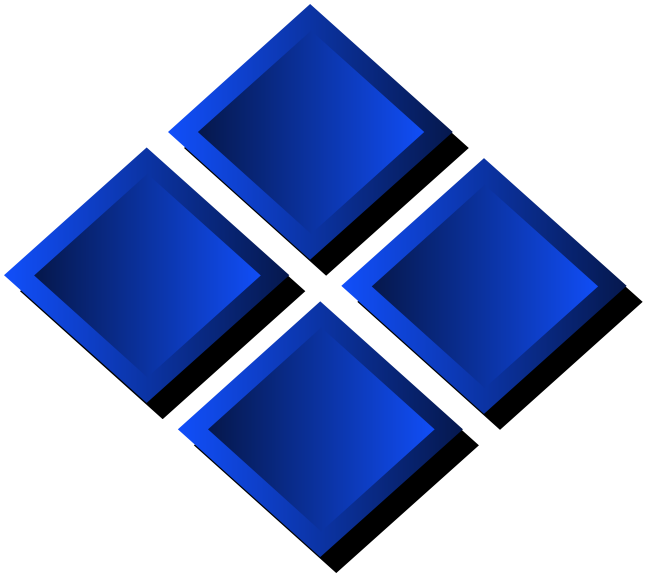
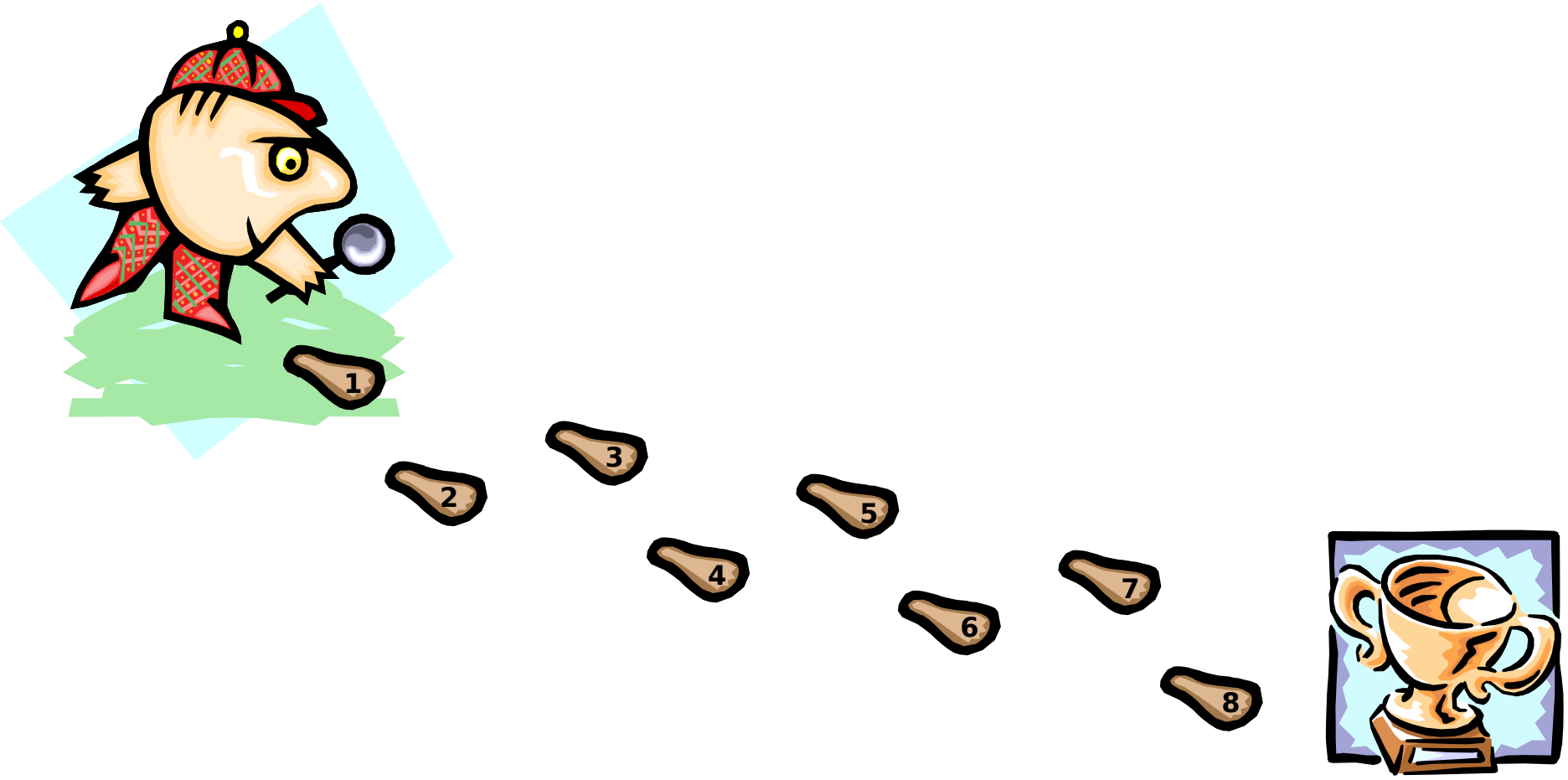


Eight Steps to Success in CMMI®-Compliant Process Engineering: Strategies and Supporting Technology

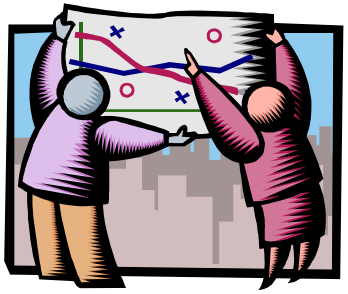
Paul R. Croll
Chair, IEEE Software Engineering Standards Committee
Vice Chair, ISO/IEC JTC1/SC7 U.S. TAG
Computer Sciences Corporation
pcroll@csc.com

A Preparatory Exercise


1
2
3
4
5
6
7
8

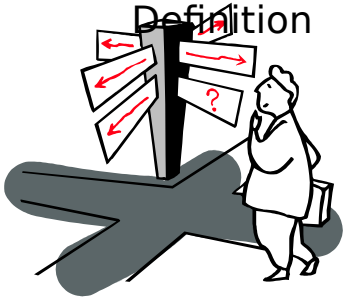
8 Steps to Success In CMMI®-Compliant Process Engineering

1. Understand your business processes



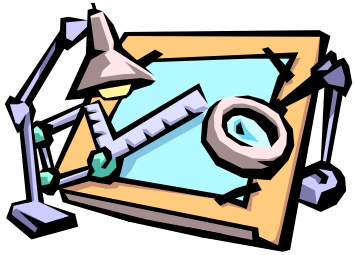
2. Look to the CMMI℠ for Process Completenes

3. Look to Framework Standards for Life Cycle Definition

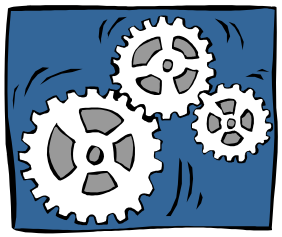


4. Look to Supporting Standards for Process Detail

5. Build or Refine Your Process Architecture



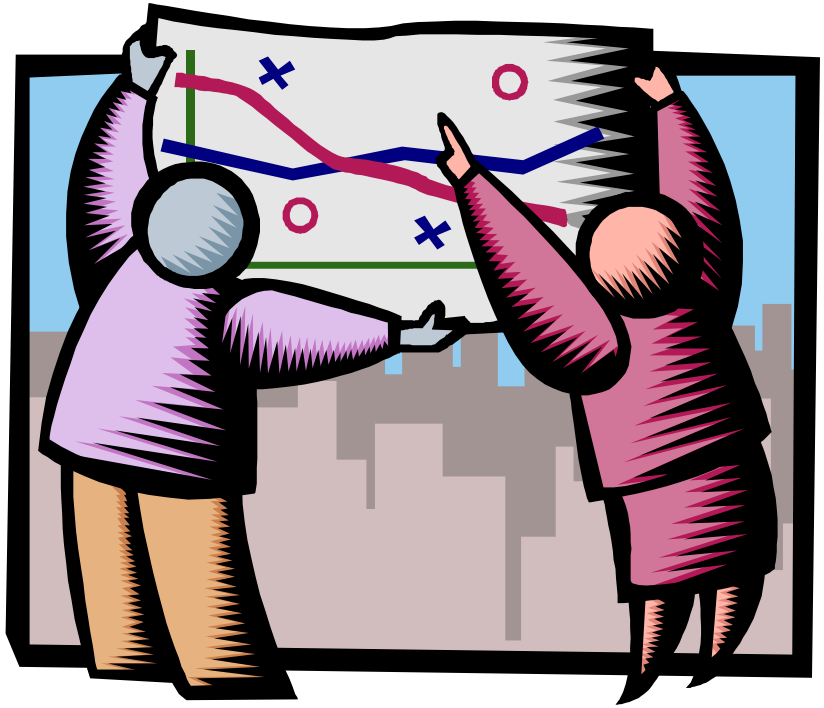
6. Execute Your Processes



7. Measure Your Results - Modify Processes as Necessary

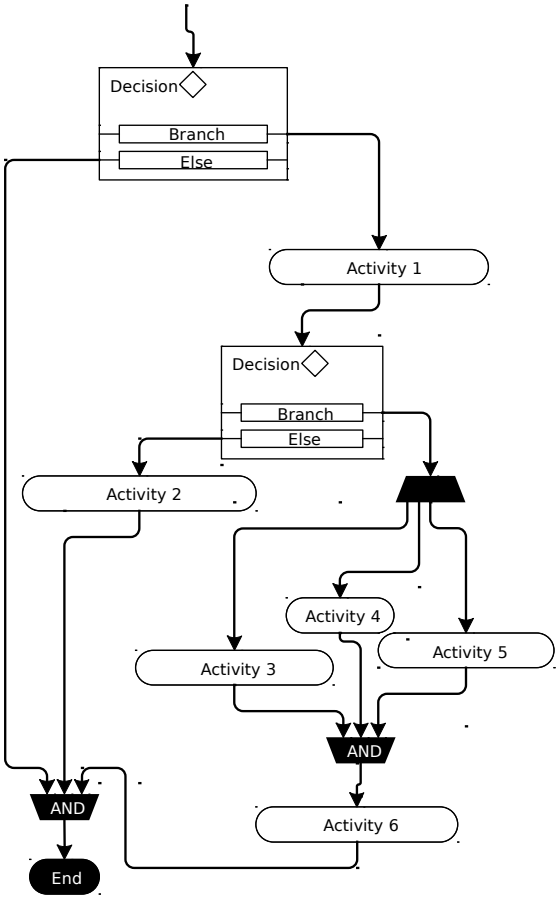
8. Confirm Your Status With Independent Appraisals

Step 1 – Understand your business processes



Strategy: Your Business is Your Business – It's Not CMMI® Implementation

- You must fully understand your business processes before you can address process completeness or process compliance.
  - What are your business processes?
  - Are they well-documented?
  - Are roles and responsibilities well-defined?
  - Are lines of authority well-defined?
  - Are internal and external interfaces well-defined?
  - Do your business processes satisfy your business goals?



Technology: Use Business Process Modeling Aids to Capture Your Business Processes

- Automated tools can help you capture and understand your business processes.
  - Business Process Management Initiative – BPML 1.0 Specification
    - http://www.bpmi.org/
  - CA's AllFusion Modeling Suite
    - http://www.asapsoftware.com/ca/allfusion.pdf
  - Casewise Corporate Modeler
    - http://www.casewise.com/
  - Popkin System Architect
    - http://www.popkin.com/
  - Proforma ProVision Modeling Suite
    - http://www.proformacorp.com/
  - Rational Suite® AnalystStudio®
    - http://www.rational.com/products/astudio/index.jsp

Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

Step 2 - Look to the CMMI® for Process Completeness

Strategy: Use the CMMI® as a Guide to Process Completeness

Determine if essential elements of your processes are missing or incomplete

Process Management

- Organizational Process Focus
- Organizational Process Definition
- Organizational Training
- Organizational Process Performance
- Organizational Innovation and Deployment

Engineering

- Requirements Management
- Requirements Development
- Technical Solution
- Product Integration
- Verification
- Validation

Project Management

- Project Planning
- Project Monitoring and Control
- Supplier Agreement Management
- Integrated Project Management for IPPD
- Risk Management
- Integrated Teaming
- Integrated Supplier Management
- Quantitative Project Management

Support

- Configuration Management
- Process and Product Quality Assurance
- Measurement and Analysis
- Decision Analysis and Resolution
- Organizational Environment for Integration
- Causal Analysis and Resolution

Technology: Use Tools and Information Assets to the Understand Your Business Processes in the CMMI® Context

- Checklists and Mapping Tables
  - http://www.stsc.hill.af.mil/consulting/cmmi/documents.html
- SEI CMMI AdoptionWeb Page
  - http://www.sei.cmu.edu/cmmi/adoption/adoption.html
- SEI Software Engineering Information Repository
  - http://seir.sei.cmu.edu/
- Basic Support for Cooperative Work (BSCW) Shared Workspace
  - http://jo.sei.cmu.edu/pub/english.cgi/0/323123
- Yahoo! CMMI Process Improvement Discussion Group
  - http://groups.yahoo.com/group/cmmi_process_improvement/

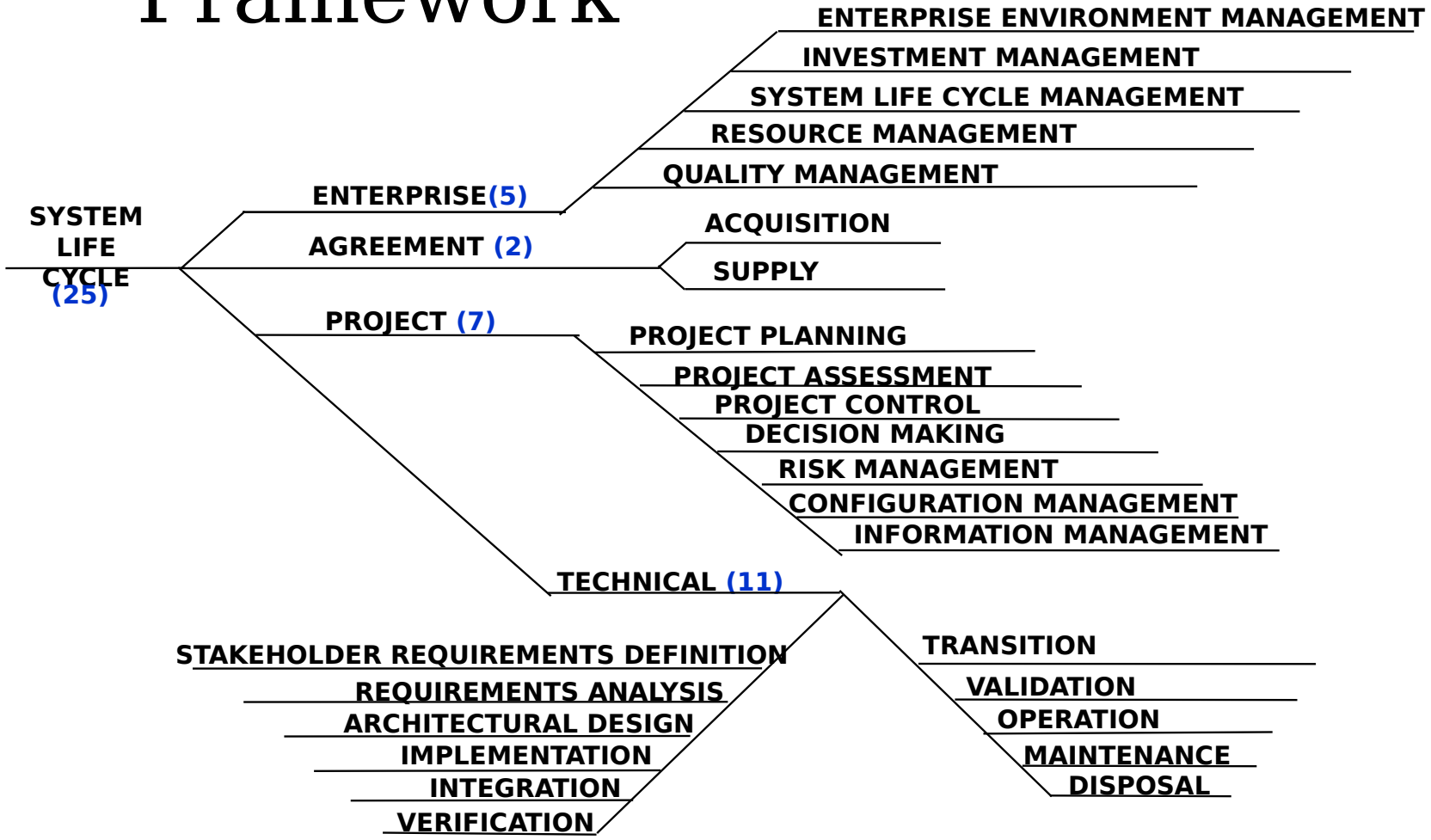
Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

Step 3 - Look to Framework Standards for Life Cycle Definition

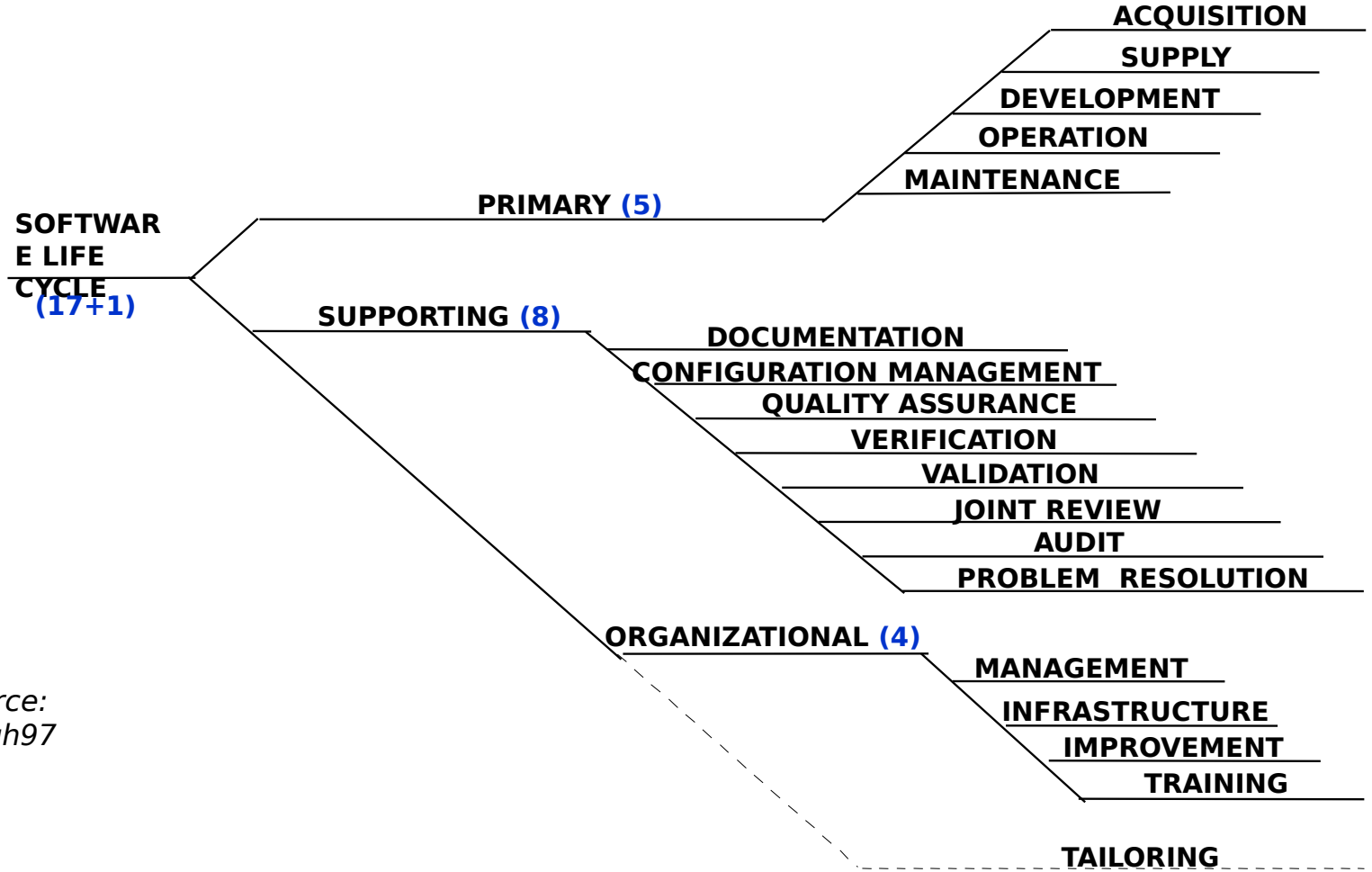
Strategy: Use Process Framework Standards to Aid in Life Cycle Definition

- Systems Life Cycle
  - **ISO/IEC 15288**, *Systems engineering — System life cycle processes*
- Software Life Cycle
  - **ISO/IEC 12207**, *Standard for Information Technology —Software life cycle processes*
  - **IEEE/EIA 12207.0**, *Industry Implementation of International Standard ISO/IEC12207:1995 — (ISO/IEC 12207) Standard for Information Technology —Software life cycle processes*
    - **IEEE/EIA 12207.1**, *Industry Implementation of International Standard ISO/IEC12207:1995 — (ISO/IEC 12207) Standard for Information Technology —Software life cycle processes – Life Cycle Data*
    - **IEEE/EIA 12207.2**, *Industry Implementation of International Standard ISO/IEC12207:1995 — (ISO/IEC 12207) Standard for Information Technology —Software life cycle processes – Implementation considerations*

The ISO/IEC 15288 Systems Life Cycle Process Framework

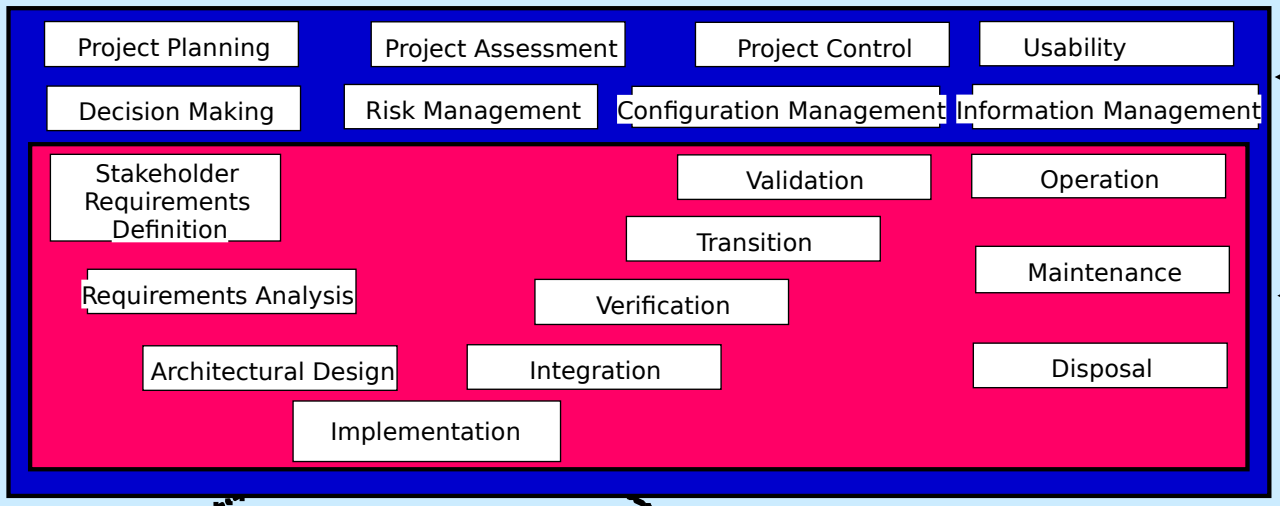


The IEEE/EIA 12207 Software Life Cycle Process Framework



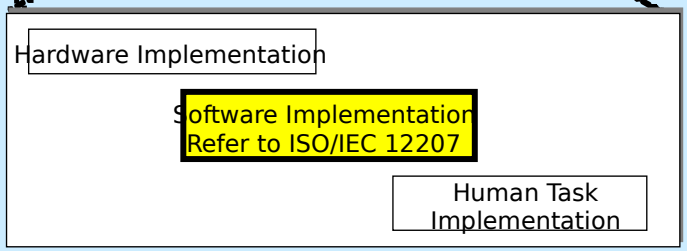
Source: Singh97

Relationship between ISO/IEC 15288 and ISO/IEC 12207

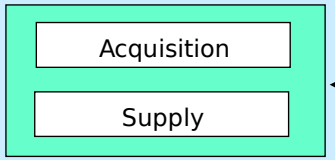


Project processes

Technical processes



Enterprise processes

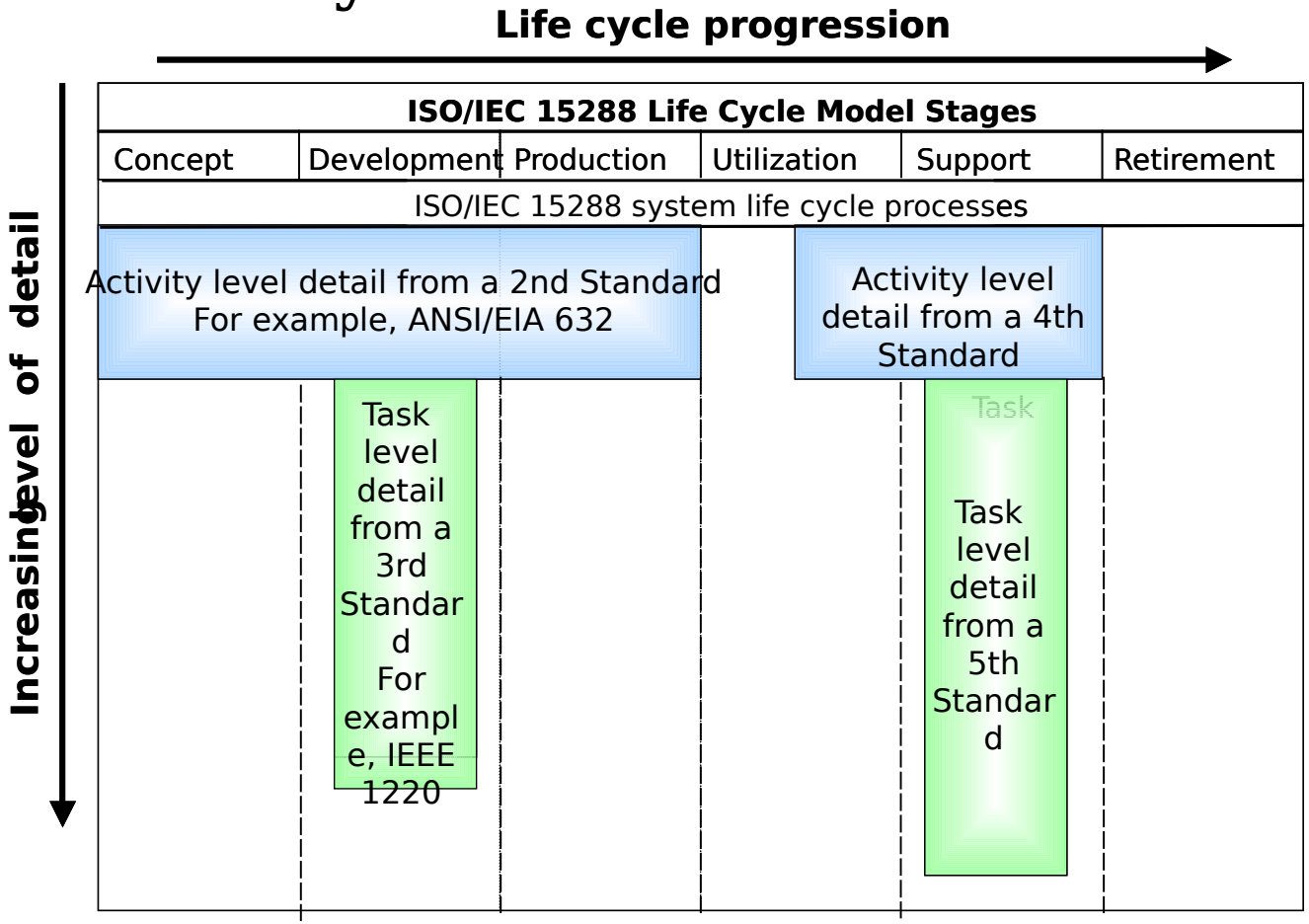


Agreement processes

Source:
ISO/IEC JTC 1/SC 7/WG 7 N0643, 2002-10-20, © ISO/IEC2002.

Step 4 – Look to Supporting Standards For Process Detail

Strategy: Use Supporting Standards as Best Practice Support for Your Defined System Life Cycle


Life cycle progression
Increasing level of detail
ISO/IEC 15288 Life Cycle Model Stages
Concept
Development
Production
Utilization
Support
Retirement
ISO/IEC 15288 system life cycle processes
Activity level detail from a 2nd Standard For example, ANSI/EIA 632
Activity level detail from a 4th Standard
Task level detail from a 3rd Standard For example, IEEE 1220
Task level detail from a 5th Standard

A1. ISO/IEC 15288 and other engineering standards

Source: Systems Engineering – A Guide for the application of ISO/IEC 15288 System Life Cycle Processes, ISO/IEC 19760, © ISO/IEC2003.

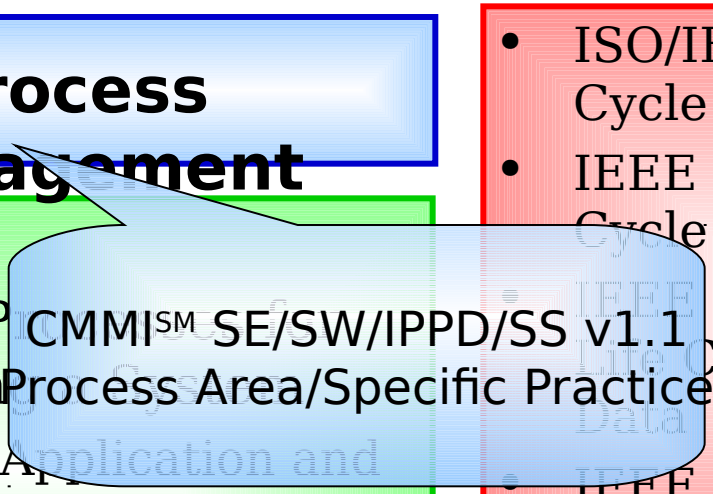
Strategy: Use Supporting Standards as Best Practice Support for Your Defined Software Life Cycle

Table 1—Information item matrix

Information item(s)	IEEE/EIA 12207.0 Clause	Kind of documentation	IEEE/EIA 12207.1 Clause	References (See annex A.)
Acceptance strategy and conditions record	5.1.1.9	Record (5.4)	—	IEEE 1062
Acquisition plan	5.1.1.8	Plan	6.1	ASTM E731, E1206, IEEE 1062
Acquisition requirements record	5.1.2.1	Record (5.4)	—	IEEE 1062, 1220
Audit agenda record	6.7.1.4	Record (5.4)	—	—
Audit procedure	6.7.1.4	Procedure (5.3)	—	—
Change request	5.4.4, 5.5.1, 6.2.3	Request	6.2	—
Concept of operations description	5.1.1.1	Description	6.3	IEEE 1362, EIA/IEEE J-STD-016 F.2.1. Also see ISO 5806, 5807, 8631, 8790, and 11411 for guidance on use of notations.
Concept/need determination record	5.1.1.1	Record (5.4)	—	IEEE 1062, 1220
Database design description	5.3.5.3, 5.3.6.3, 5.3.7.1	Description	6.4	IEEE 1016, EIA/IEEE J-STD-016 G.2.3
Detailed design evaluation record	5.3.6.7	Record	6.6	—
Development process plan	5.3.1.4	Plan	6.5	ASTM E622, E1340, EIA/IEEE J-STD-016 E.2.1, IEEE 1074, 1074.1

Source: IEEE/EIA 12207.1-1997, © IEEE 1998.

CMMI® SE/SW/IPPD/SS v1.1 Standards Mapping - Process Management



Process Management

- EIA 632 - Processes for Engineering a System
- IEEE 1220, Application and Management of the Systems Engineering Process
- IEEE 12207, Standard for Software Life Cycle Processes
- 1517-1999, Reuse Processes

Framework Standards

- ISO/IEC 15288, System Life Cycle Processes
- IEEE 12207.0, Software Life Cycle Processes
- IEEE 12207.1, Guide to Software Life Cycle Processes—Life Cycle Data
- IEEE 12207.2, Guide to Software Life Cycle Processes—Implementation Considerations

Supporting Standards

CMMI® SE/SW/IPPD/SS v1.1 Standards Mapping – Project Management

Project Management

- IEEE 1220, Application and Management of the Systems Engineering Process
- IEEE 1058, Software Project Management Plans
- IEEE 1490, A Guide to the Program Management Body of Knowledge
- IEEE 1062, Recommended Practice for Software Acquisition
- IEEE 1540, Risk Management
- IEEE 1028, Software Reviews
- ISO/IEC 15288, System Life Cycle Processes
- IEEE 12207.0, Software Life Cycle Processes
- IEEE 12207.1, Guide to Software Life Cycle Processes—Life Cycle Data
- IEEE 12207.2, Guide to Software Life Cycle Processes—Implementation Considerations

CMMI® SE/SW/IPPD/SS v1.1 Standards Mapping – Engineering

Engineering

- IEEE 1233, Guide for Developing System Requirements Specifications
- IEEE 1362, Guide for Concept of Operations Document
- IEEE 1471, Architectural Description of Software Intensive Systems
- IEEE 830, Software Requirements Specifications
- IEEE 1016, Software Design Descriptions
- IEEE 1012, Software Verification and Validation
- IEEE 1008, Software Unit Testing

- ISO/IEC 15288, System Life Cycle Processes
- IEEE 12207.0, Software Life Cycle Processes
- IEEE 12207.1, Guide to Software Life Cycle Processes—Life Cycle Data
- IEEE 12207.2, Guide to Software Life Cycle Processes—Implementation Considerations

- IEEE 1228, Software Safety Plans
- IEEE 1063, Software User Documentation
- IEEE 1219, Software Maintenance
- IEEE 1320.1,.2, IDEF0, IDEF1X97
- IEEE 1420.1, Data Model for Reuse Library Interoperability

CMMI® SE/SW/IPPD/SS v1.1 Standards Mapping – Support

Support

- IEEE 828, Software Configuration Management Plans
- IEEE 730, Software Quality Assurance Plans
- IEEE 982.1, Dictionary of Measures to Produce Reliable Software
- IEEE 1045, Software Productivity Metrics
- IEEE 1061, Software Quality Metrics Methodology
- IEEE 1219, Software Maintenance

- ISO/IEC 15288, System Life Cycle Processes
- IEEE 12207.0, Software Life Cycle Processes
- IEEE 12207.1, Guide to Software Life Cycle Processes—Life Cycle Data
- IEEE 12207.2, Guide to Software Life Cycle Processes—Implementation Considerations

- IEEE 1465 (ISO/IEC 12119) - Software Packages - Quality Requirements and Testing
- IEEE 14143.1 (ISO/IEC 1443-1) - Functional Size Measurement - Part 1: Definition of Concepts

An Example - Requirements Development

SP 2.1-1 Establish Product and Product Component Requirements

- Establish and maintain, from the customer requirements, product and product component requirements essential to product and product component effectiveness and affordability

- ISO/IEC 15288, System Life Cycle Processes
  - Clause 5.5.3 - Requirements Analysis Process
- IEEE/EIA 12207.0, Software Life Cycle Processes
  - Clause 5.3.2 - System Requirements Analysis
  - Clause 5.3.4 - Software requirements analysis

- IEEE 1233, Guide for Developing System Requirements Specifications
- IEEE 830, Software Requirements Specifications

An Example - Requirements Development

SP 2.1-1 Establish Product and Product Component Requirements

- ISO/IEC 15288, System Life Cycle Processes
  - Clause 5.5.3 - Requirements Analysis Process
  - IA 12207.0, Software Life rocesses
  - se 5.3.2 - System uirements Analysis
  - se 5.3.4 - Software uirements analysis

233, Guide for Developing Requirements cations

30, Software ements Specifications

5.5.3 Requirements Analysis Process
5.5.3.1 Purpose of the Requirements Analysis Process
The purpose of the Requirements Analysis Process is to transform the stakeholder, requirement-driven view of desired services into a technical view of a required product that could deliver those services.
This process builds a representation of a future system that will meet stakeholder requirements and that, as far as constraints permit, does not imply any specific implementation. It results in measurable system requirements that specify, from the developer's perspective, what characteristics it is to possess and with what magnitude in order to satisfy stakeholder requirements.
5.5.3.2 Requirements Analysis Process Outcomes
As a result of the successful implementation of the Requirements Analysis Process:
a) The required characteristics, attributes, and functional and performance requirements for a product solution are specified.
b) Constraints that will affect the architectural design of a system and the means to realize it are specified.
c) The integrity and traceability of system requirements to

Source: ISO/IEC CD 15288 FDIS, © ISO/IEC2002.

An Example - [illegible]s

5.3.2.1 The specific intended use of the system to be developed shall be analyzed to specify system requirements. The system requirements specification shall describe: functions and capabilities of the system; business, organizational and user requirements; safety, security, human-factors engineering (ergonomics), interface, operations, and maintenance requirements; design constraints and qualification requirements. The system requirements specification shall be documented.
5.3.4.1 The developer shall establish and document software requirements, including the quality characteristics specifications, described below. . . .
a) Functional and capability specifications, including performance, physical characteristics, and environmental conditions under which the software item is to perform;
b) Interfaces external to the software item;
c) Qualification requirements;
d) Safety specifications, including those related to methods of operation and maintenance, environmental influences, and personnel injury;
e) Security specifications, including those related to compromise of sensitive information . . .

Source: IEEE/EIA 12207.0-1997, © IEEE 2001.

SO/IEC 15288, System Life Cycle Processes

- Clause 5.5.3 - Requirements Analysis Process

EE/EIA 12207.0, Software Life ycle Processes

- Clause 5.3.2 - System Requirements Analysis
- Clause 5.3.4 - Software requirements analysis

- IEEE 1233, Guide for Developing System Requirements Specifications
- IEEE 830, Software Requirements Specifications

An Example -

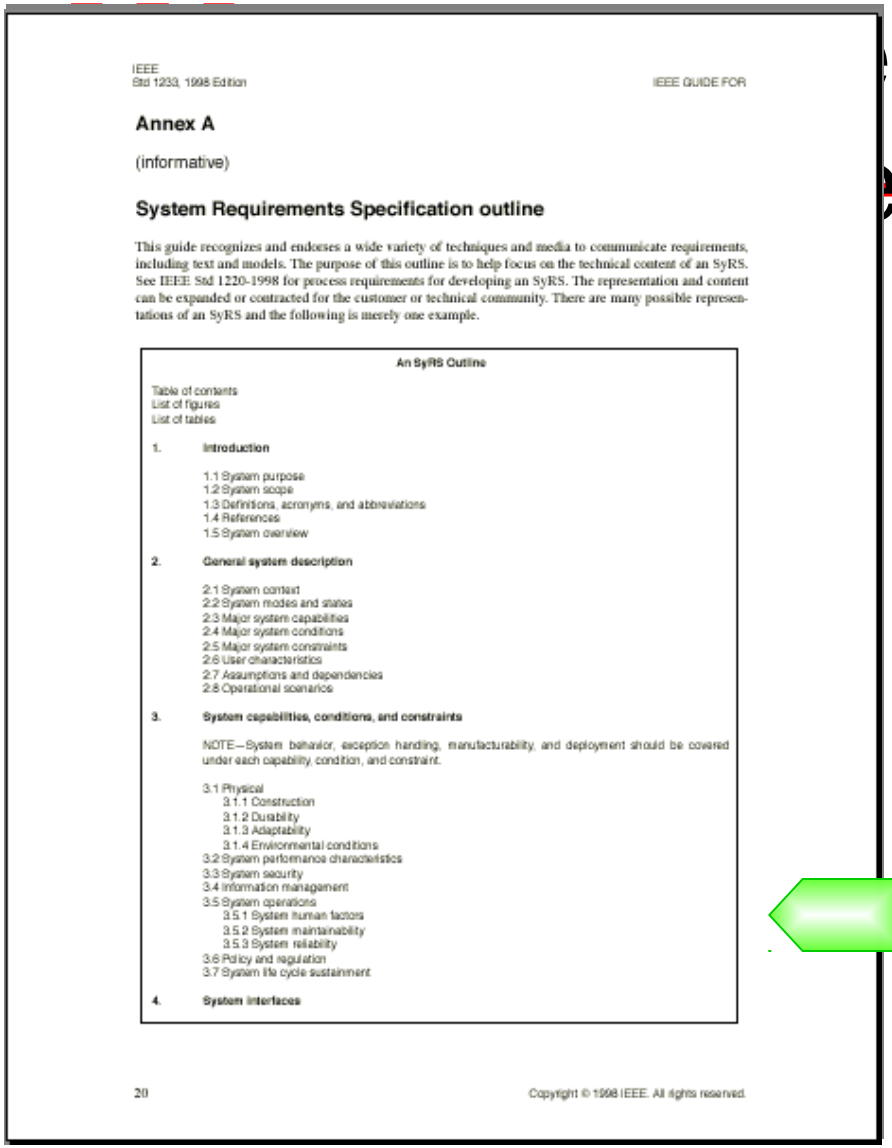
7.2 Build a well-formed requirement
The analysts carry out this subphase by doing the following:
a) Ensuring that each requirement is a necessary, short, definitive statement of need (capability, constraints);
b) Defining the appropriate conditions (quantitative or qualitative measures) for each requirement and avoiding adjectives such as “resistant” or “industry wide;”
c) Avoiding requirements pitfalls (see 6.4);
d) Ensuring the readability of requirements, which entails the following:
1) Simple words/phrases/concepts;
2) Uniform arrangement and relationship;
3) Definition of unique words, symbols, and notations;
4) The use of grammatically correct language and symbology.
e) Ensuring testability.
Example:
Capability: Move people between Los Angeles and New York
Condition: Cruising speed of 200 km/hr
Constraint: Maximum speed of 300 km/hr
Well-formed requirement: This system should move people between Los Angeles and New York at an optimal cruising speed of 200 km/hr with a maximum speed of 300 km/hr.

Source: IEEE 1233-1998, © IEEE 1998.

IEC 15288, System Life
e Processes
Clause 5.5.3 - Requirements
Analysis Process
E/EIA 12207.0, Software Life
e Processes
Clause 5.3.2 - System
Requirements Analysis
Clause 5.3.4 - Software
Requirements analysis

- IEEE 1233, Guide for Developing System Requirements Specifications
- IEEE 830, Software Requirements Specifications

An Example - nts ent



IEEE
Std 1233, 1998 Edition

IEEE GUIDE FOR

Annex A

(informative)

System Requirements Specification outline

This guide recognizes and endorses a wide variety of techniques and media to communicate requirements, including text and models. The purpose of this outline is to help focus on the technical content of an SyRS. See IEEE Std 1220-1998 for process requirements for developing an SyRS. The representation and content can be expanded or contracted for the customer or technical community. There are many possible representations of an SyRS and the following is merely one example.

An SyRS Outline

Table of contents
List of figures
List of tables

1. **Introduction**

1.1 System purpose
1.2 System scope
1.3 Definitions, acronyms, and abbreviations
1.4 References
1.5 System overview

2. **General system description**

2.1 System context
2.2 System modes and states
2.3 Major system capabilities
2.4 Major system conditions
2.5 Major system constraints
2.6 User characteristics
2.7 Assumptions and dependencies
2.8 Operational scenarios

3. **System capabilities, conditions, and constraints**

NOTE—System behavior, exception handling, manufacturability, and deployment should be covered under each capability, condition, and constraint.

3.1 Physical
3.1.1 Construction
3.1.2 Durability
3.1.3 Adaptability
3.1.4 Environmental conditions
3.2 System performance characteristics
3.3 System security
3.4 Information management
3.5 System operations
3.5.1 System human factors
3.5.2 System maintainability
3.5.3 System reliability
3.6 Policy and regulation
3.7 System life cycle sustainment

4. **System interfaces**

20

Copyright © 1998 IEEE. All rights reserved.

Source: IEEE 1233-1998, © IEEE 1998.

- ISO/IEC 15288, System Life Cycle Processes
  - Clause 5.5.3 - Requirements Analysis Process
- IEEE/EIA 12207.0, Software Life Cycle Processes
  - Clause 5.3.2 - System Requirements Analysis
  - Clause 5.3.4 - Software requirements analysis

- IEEE 1233, Guide for Developing System Requirements Specifications
- IEEE 830, Software Requirements Specifications

An Example

5.3.2 Functions
Functional requirements should define the fundamental actions that must take place in the software in accepting and processing the inputs and in processing and generating the outputs. These are generally listed as “shall” statements starting with “The system shall”
These include:
a) Validity checks on the inputs
b) Exact sequence of operations
c) Responses to abnormal situations, including:
  1) Overflow
  2) Communication facilities
  3) Error handling and recovery
d) Effect of parameters
e) Relationship of outputs to inputs . . .
  1) It may be appropriate to partition the functional requirements into subfunctions or subprocesses. This does not imply that the software design will also be partitioned that way.
5.3.3 Performance requirements
This subsection should specify both the static and the dynamic numerical requirements placed on the software or on human interaction with the software as a whole. Static numerical requirements may include the following:
a) The number of terminals to be supported;
b) The number of simultaneous users to be supported;
c) Amount and type of information to be handled.

Source:
IEEE 830-1998, © IEEE 1998.

IEC 15288, System Life e Processes

Clause 5.5.3 - Requirements Analysis Process

E/EIA 12207.0, Software Life e Processes

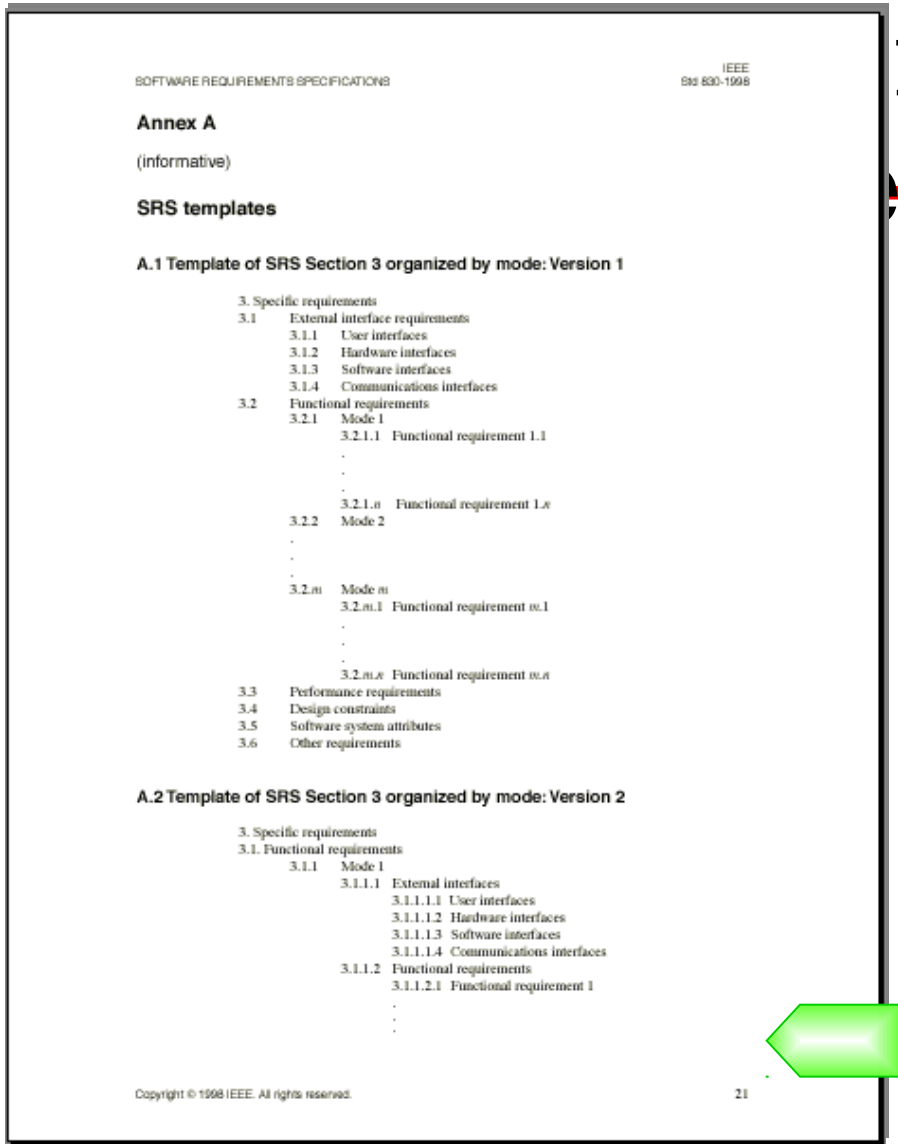
Clause 5.3.2 - System Requirements Analysis

Clause 5.3.4 - Software equirements analysis

EE 1233, Guide for Developing ystem Requirements Specifications

- IEEE 830, Software Requirements Specifications

An Example - nts nt



SOFTWARE REQUIREMENTS SPECIFICATIONS — IEEE Std 830-1998

Annex A

(informative)

SRS templates

A.1 Template of SRS Section 3 organized by mode: Version 1

3. Specific requirements
3.1 External interface requirements
3.1.1 User interfaces
3.1.2 Hardware interfaces
3.1.3 Software interfaces
3.1.4 Communications interfaces
3.2 Functional requirements
3.2.1 Mode 1
3.2.1.1 Functional requirement 1.1
.
.
.
3.2.1.*n* Functional requirement 1.*n*
3.2.2 Mode 2
.
.
.
3.2.*m* Mode *m*
3.2.*m*.1 Functional requirement *m*.1
.
.
.
3.2.*m*.*n* Functional requirement *m*.*n*
3.3 Performance requirements
3.4 Design constraints
3.5 Software system attributes
3.6 Other requirements

A.2 Template of SRS Section 3 organized by mode: Version 2

3. Specific requirements
3.1. Functional requirements
3.1.1 Mode 1
3.1.1.1 External interfaces
3.1.1.1.1 User interfaces
3.1.1.1.2 Hardware interfaces
3.1.1.1.3 Software interfaces
3.1.1.1.4 Communications interfaces
3.1.1.2 Functional requirements
3.1.1.2.1 Functional requirement 1
.
.
.

Copyright © 1998 IEEE. All rights reserved.

Source: IEEE 830-1998, © IEEE 1998.

- ISO/IEC 15288, System Life Cycle Processes
  - Clause 5.5.3 - Requirements Analysis Process
- IEEE/EIA 12207.0, Software Life Cycle Processes
  - Clause 5.3.2 - System Requirements Analysis
  - Clause 5.3.4 - Software requirements analysis

- IEEE 1233, Guide for Developing System Requirements Specifications
- IEEE 830, Software Requirements Specifications

Technology: Use Mapping Tools and Reports to Select Supporting Standards and Identify Best Practice Detail

- The IEEE CMMI® Mapping Project
  - http://computer.org/standards/sesc/
- The FAA-iCMM® Project
  - http://www2.faa.gov/aio/ProcessEngr/iCMM/index.htm
- The Software Productivity Consortium's Quagmap® Tool
  - http://www.software.org/pub/products/quagmap.asp

Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

The IEEE CMMI® Mapping Project – Background

- Study Group on External Standards Coordination formed
  - Initial Task: Map the IEEE Software Engineering Standards Collection to the CMMI®.
- Software Productivity Consortium's Quagmap® selected as the mapping tool.
- License agreements for distribution of Standards and the Quagmap® tool to Study Group members negotiated and signed.
- Results to be published on the SESC web site.

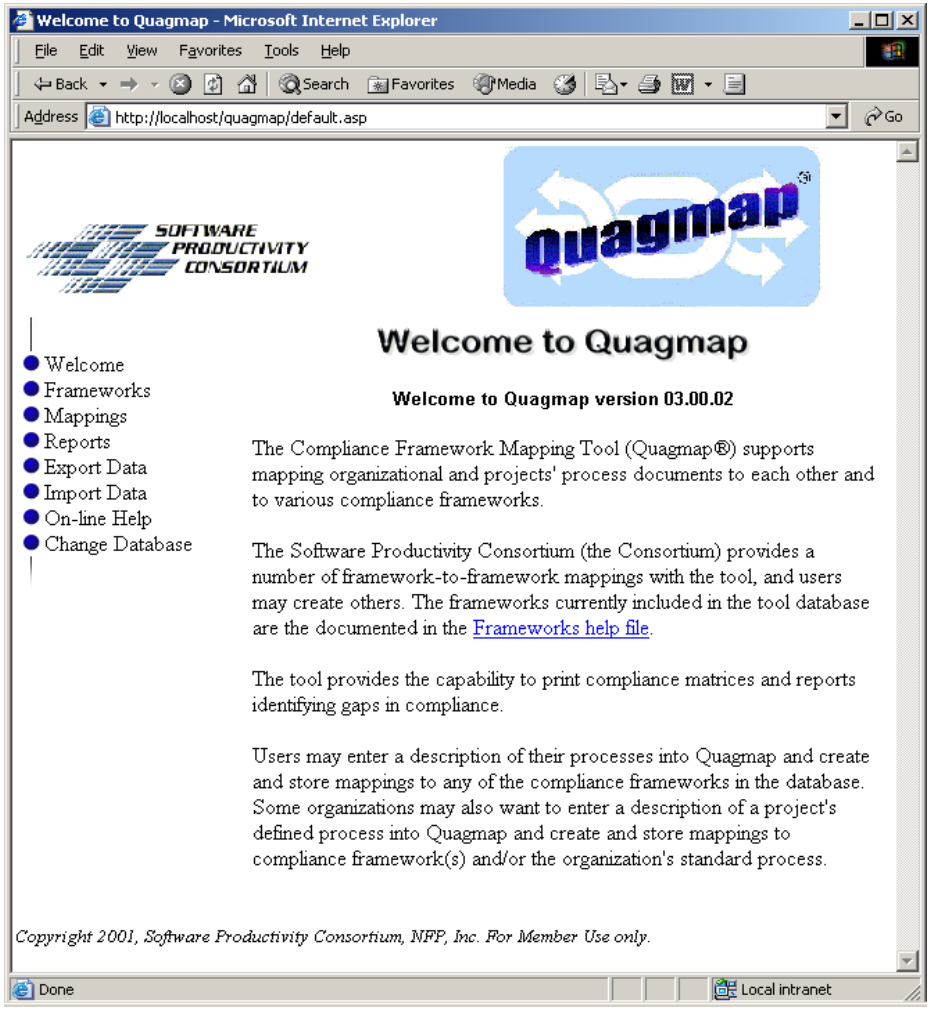
Best Practice Information Aids For CMMI®-Compliant Process Engineering

610.12-1990	IEEE Standard Glossary of Software Engineering Terminology
730-1998	IEEE Standard for Software Quality Assurance Plans
828-1998	IEEE Standard for Software Configuration Management Plans
829-1998	IEEE Standard for Software Test Documentation
830-1998	IEEE Recommended Practice for Software Requirements Specifications
982.1-1988	IEEE Standard Dictionary of Measures to Produce Reliable Software
1008-1987	IEEE Standard for Software Unit Testing
1012-1998	IEEE Standard for Software Verification and Validation
1012a-1998	IEEE Standard for Software Verification and Validation - Supplement to 1012-1998 - Content Map to IEEE 12207.1
1016-1998	IEEE Recommended Practice for Software Design Descriptions
1028-1997	IEEE Standard for Software Reviews
1044-1993	IEEE Standard Classification for Software Anomalies
1045-1992	IEEE Standard for Software Productivity Metrics
1058-1998	IEEE Standard for Software Project Management Plans
1058.1-1987	IEEE Standard for Software Project Management Plans
1061-1998	IEEE Standard for Software Quality Metrics Methodology
1062, 1998	IEEE Recommended Practice for Software Acquisition (includes IEEE 1062a)
1063-2001	IEEE Standard for Software User Documentation
1074-1997	IEEE Standard for Developing Software Life Cycle Processes
1175.1-2002	IEEE Guide for CASE Tool Interconnections-Classification and Description
1219-1998	IEEE Standard for Software Maintenance
1220-1998	IEEE Standard for the Application and Management of the Systems Engineering Process
1228-1994	IEEE Standard for Software Safety Plans
1233-1998	IEEE Guide for Developing System Requirements Specifications (including IEEE 1233a)
1320.1-1998	IEEE Standard for Functional Modeling Language - Syntax and Semantics for IDEF0
1320.2-1998	IEEE Standard for Conceptual Modeling Language Syntax and Semantics for IDEF1X97 (IDEFobject)
1362-1998	IEEE Guide for Information Technology-System Definition -Concept of Operation Document
1420.1-1995	IEEE Standard for Information Technology--Software Reuse--Data Model for Reuse Library Interoperability: Basic Interoperability Data Model (BIDM)
1420.1a-1996	IEEE Supplement to Standard for Information Technology--Software Reuse--Data Model for Reuse Library Interoperability: Asset Certification Framework
1420.1b-1999	IEEE Trial-use Supplement to IEEE Standard for Information Technology--Software Reuse--Data Model for Reuse Library Interoperability: Intellectual Property Rights Framework
1465-1998	(12119:1998 ISO/IEC) Information Technology - Software Packages - Quality Requirements and Testing
1471-2000	IEEE Recommended Practice for Architectural Description of Software Incentive Systems
1490-1998	IEEE Guide (©IEEE) - Adoption of PMI Standard - A Guide to the Project Management Body of Knowledge (©PMI)
1517-1999	IEEE Standard for Information Technology - Software Life Cycle Processes - Reuse Processes
1540-2001	IEEE Standard for Software Life Cycle Processes-Risk Management
2001-1999	IEEE Recommended Practice for Internet Practices--Web page Engineering--Intranet/Extranet Applications
12207.0-1996	IEEE/EIA Standard: Industry Implementation of International Standard ISO/IEC 12207:1995 Standard for Information Technology--Software Life Cycle Processes
12207.1-1997	IEEE/EIA Standard: Industry Implementation of International Standard ISO/IEC 12207:1995 Standard for Information Technology-- Software Life Cycle Processes--Life cycle data
12207.2-1997	IEEE/EIA Standard: Industry Implementation of International Standard ISO/IEC 12207:1995 Standard for Information Technology-- Software Life Cycle Processes--Implementation considerations
14143.1-2000	Implementation Note for IEEE Adoption of ISO/IEC 14143-1:1998, Information Technology-Software Measurement-Functional Size Measurement-Part 1: Definition of Concepts

℠ CMMI is a service mark of Carnegie Mellon University

For more information: http://standards.computer.org/sesc/

The SPC Quagmap® Tool


Welcome to Quagmap - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/quagmap/default.asp
Go
SOFTWARE PRODUCTIVITY CONSORTIUM
quagmap
Welcome
Frameworks
Mappings
Reports
Export Data
Import Data
On-line Help
Change Database
Welcome to Quagmap
Welcome to Quagmap version 03.00.02
The Compliance Framework Mapping Tool (Quagmap®) supports mapping organizational and projects' process documents to each other and to various compliance frameworks.
The Software Productivity Consortium (the Consortium) provides a number of framework-to-framework mappings with the tool, and users may create others. The frameworks currently included in the tool database are the documented in the Frameworks help file.
The tool provides the capability to print compliance matrices and reports identifying gaps in compliance.
Users may enter a description of their processes into Quagmap and create and store mappings to any of the compliance frameworks in the database. Some organizations may also want to enter a description of a project's defined process into Quagmap and create and store mappings to compliance framework(s) and/or the organization's standard process.
Copyright 2001, Software Productivity Consortium, NFP, Inc. For Member Use only.
Done
Local intranet

Source: Software Productivity Consortium www.software.org

Quagmap® Mappings and Relationships

- **Mappings** define how well one framework (the Source) meets the intent of the second (the Target)
- **Relationships** define how well each source framework data item meets the intent of the associated target framework item[s]
  - [none]
  - Partial
  - Complete
- **Standards mappings** are generally done in both directions

Source: Software Productivity Consortium, www.software.org

Quagmap® Reports

- **Framework:** Description and all data items
- **Relationship:** Source framework items and related data items from one or more target frameworks
- **Coverage:** Target framework items and the source data items that meet them
- **Gap Identification:** Target items not fully satisfied by the source data items
- **Inferred Mapping:** Given a mapping from A:B, and B:C; what is a likely set of relationships from A:C?

Source: Software Productivity Consortium, www.software.org

Quagmap® Relationship Report

Relationship Report - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Relationship Report

Source: IEEE/EIA 12207 (1996)
Notes:

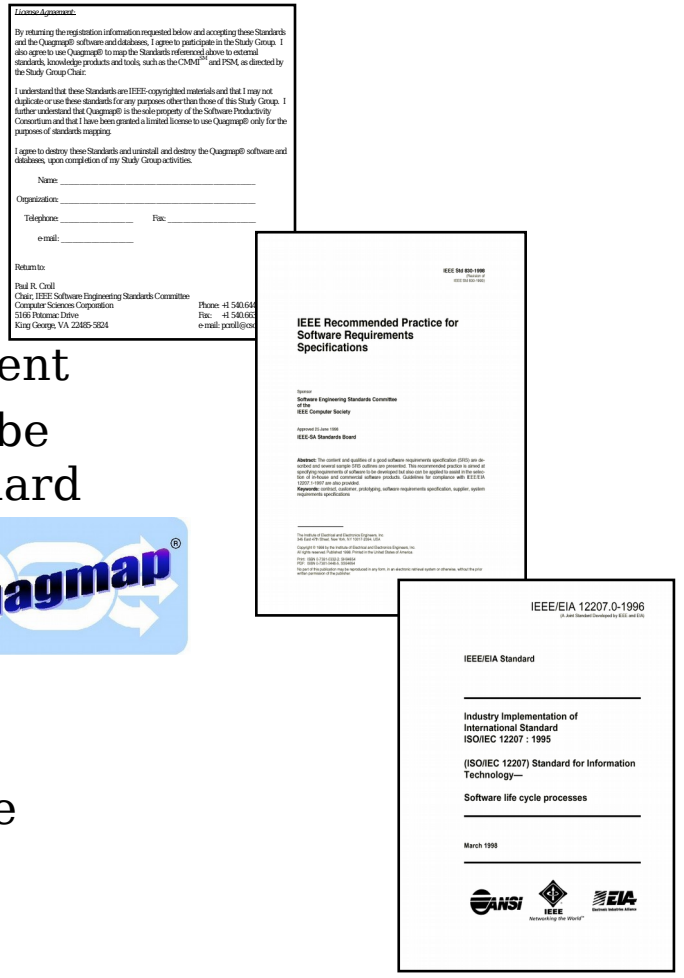
Printed: 11/12/2002
Format to Save or Print

Source Item Name	Source Item Description	Target Framework	Target Item Name	Target Item Description	Relationship	Comment
				according to a documented procedure		
5.1.1.9	Initiation, Task 9	SW-CMM V1.1 (1.1)	SSM-AC-3	The contractual agreement between the prime contractor and the software subcontractor is used as the basis for managing the subcontract	Partial	
5.1.1.9	Initiation, Task 9	SW-CMM V1.1 (1.1)	SSM-AC-12	The prime contractor conducts acceptance testing as part of the delivery of the subcontractor's software products according to a documented procedure	Partial	
5.1.2.1	Request-for-proposal [-tender] preparation, Task 1	SW-CMM V1.1 (1.1)	SSM-AC-1	The work to be subcontracted is defined and planned according to a documented procedure	Partial	
5.1.2.3	Request-for-proposal [-tender] preparation, Task 3	SW-CMM V1.1 (1.1)	SSM-AC-3	The contractual agreement between the prime contractor and the software subcontractor is used as the basis for managing the subcontract	Partial	
5.1.2.4	Request-for-proposal [-	SW-CMM V1.1	SSM-AC-1	The work to be	Partial	

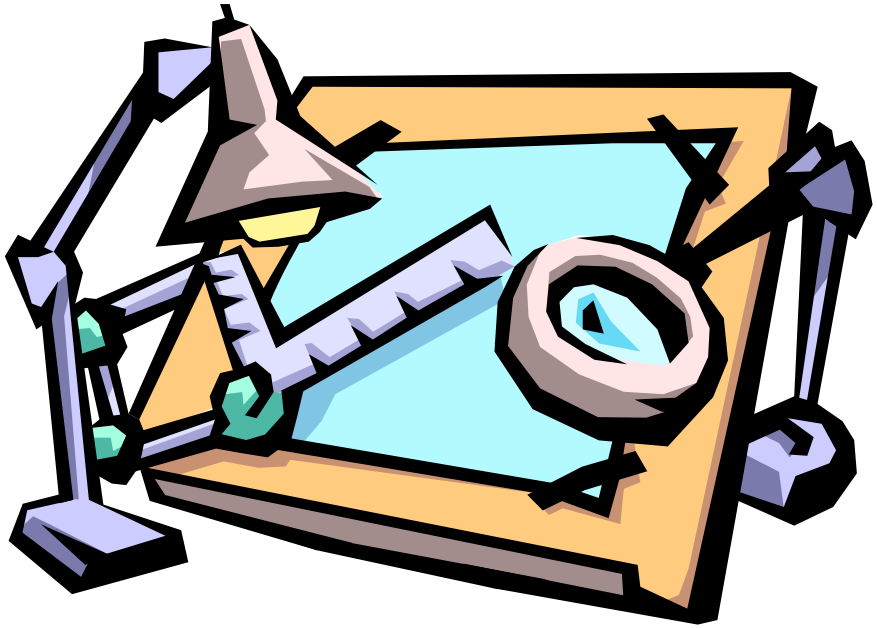
Source: Software Productivity Consortium www.software.org

How You Can Help . . .

- Get involved in the mapping effort
  - Initial mapping
  - Peer reviewer
- What you will receive
  - A Copyright Notice and License Agreement
  - An electronic copy of the standard(s) to be mapped the associated framework standard (s)
  - A copy of the Quagmap® tool
  - A mapping guide
- How to get started
  - Contact me at: pcroll@csc.com to let me know of your interest


License Agreement
IEEE Recommended Practice for Software Requirements Specifications
IEEE/EIA 12207.0-1996
IEEE/EIA Standard
Industry Implementation of International Standard ISO/IEC 12207 : 1995
(ISO/IEC 12207) Standard for Information Technology—
Software life cycle processes
March 1998

Step 5 – Build or Refine Your Process Architecture



Strategy: Use Best Practice Information to Build or Refine Your Process Architecture

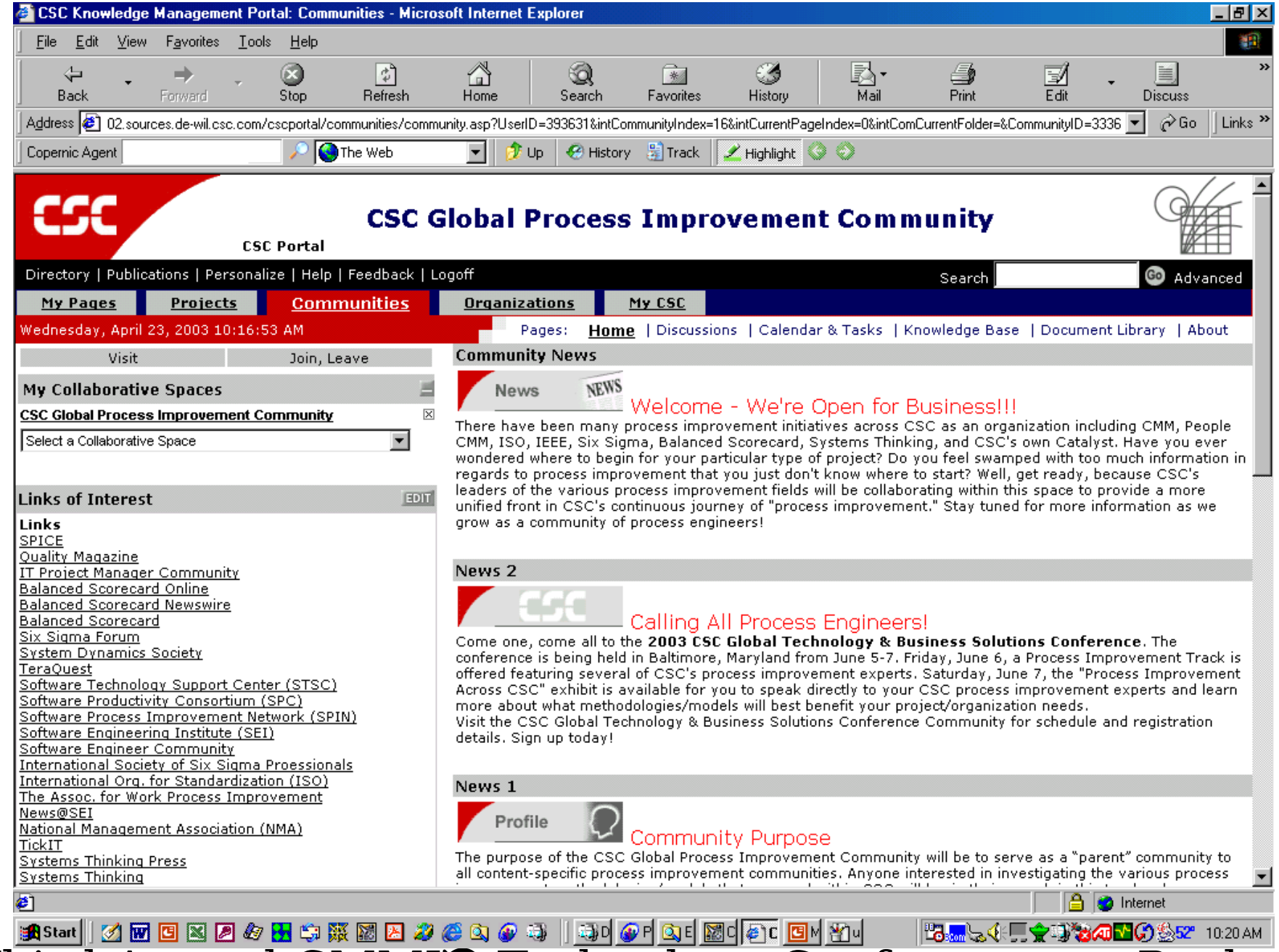
- Existing process assets
- CMMI℠ gap identification
- Framework and Supporting Standards
- Industry and DoD best practices
  - SEI CMMI AdoptionWeb Page
    - http://www.sei.cmu.edu/cmmi/adoption/adoption.html
  - SEI Software Engineering Information Repository
    - http://seir.sei.cmu.edu/
  - USAF Software Technology Support Center
    - http://www.stsc.hill.af.mil
  - Navy SPAWAR Systems Engineering Process Office
    - http://sepo.spawar.navy.mil/sepo/index2.html

Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

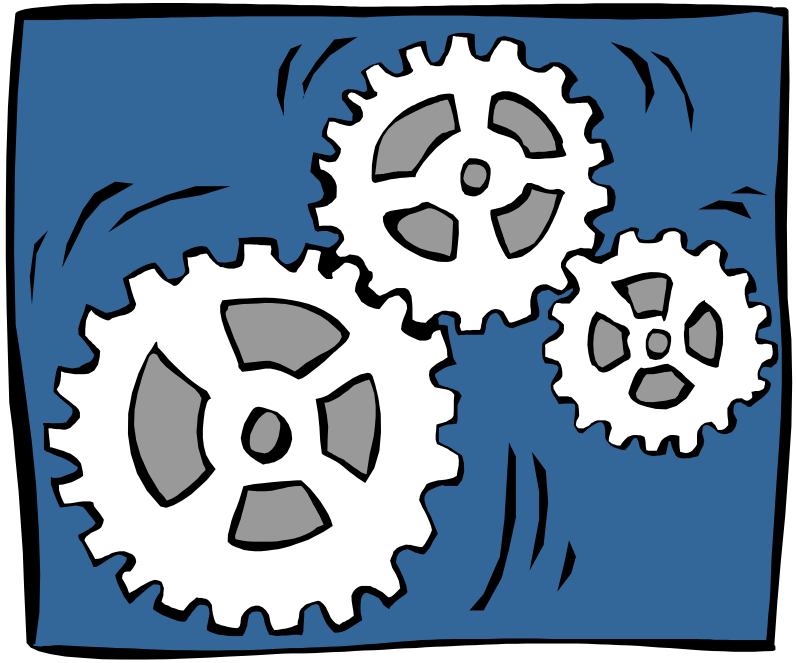
Technology: Use Tools to Support Process Definition

- CMMI® Reference Card
  - http://www.sei.cmu.edu/cmmi/publications/ref-card.pdf
- Process Asset Libraries
- Knowledge Portals
- pragma Systems Corporation's processMax5
  - http://www.pragmasystems.com/max5fin.htm

Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

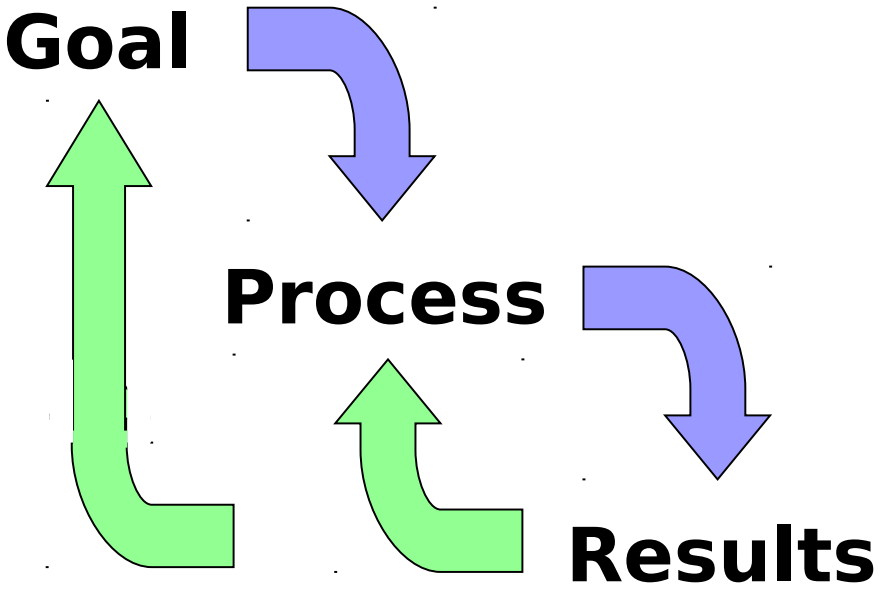

CSC Knowledge Management Portal: Communities - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss
Address 02.sources.de-wil.csc.com/cscportal/communities/community.asp?UserID=393631&intCommunityIndex=16&intCurrentPageIndex=0&intComCurrentFolder=&CommunityID=3336
Go Links
Copernic Agent The Web Up History Track Highlight
CSC
CSC Portal
CSC Global Process Improvement Community
Directory | Publications | Personalize | Help | Feedback | Logoff
Search Go Advanced
My Pages Projects Communities Organizations My CSC
Wednesday, April 23, 2003 10:16:53 AM
Pages: Home | Discussions | Calendar & Tasks | Knowledge Base | Document Library | About
Visit Join, Leave
My Collaborative Spaces
CSC Global Process Improvement Community
Select a Collaborative Space
Links of Interest
EDIT
Links
SPICE
Quality Magazine
IT Project Manager Community
Balanced Scorecard Online
Balanced Scorecard Newswire
Balanced Scorecard
Six Sigma Forum
System Dynamics Society
TeraQuest
Software Technology Support Center (STSC)
Software Productivity Consortium (SPC)
Software Process Improvement Network (SPIN)
Software Engineering Institute (SEI)
Software Engineer Community
International Society of Six Sigma Proessionals
International Org. for Standardization (ISO)
The Assoc. for Work Process Improvement
News@SEI
National Management Association (NMA)
TickIT
Systems Thinking Press
Systems Thinking
Community News
News
NEWS
Welcome - We're Open for Business!!!
There have been many process improvement initiatives across CSC as an organization including CMM, People CMM, ISO, IEEE, Six Sigma, Balanced Scorecard, Systems Thinking, and CSC's own Catalyst. Have you ever wondered where to begin for your particular type of project? Do you feel swamped with too much information in regards to process improvement that you just don't know where to start? Well, get ready, because CSC's leaders of the various process improvement fields will be collaborating within this space to provide a more unified front in CSC's continuous journey of "process improvement." Stay tuned for more information as we grow as a community of process engineers!
News 2
CSC
Calling All Process Engineers!
Come one, come all to the 2003 CSC Global Technology & Business Solutions Conference. The conference is being held in Baltimore, Maryland from June 5-7. Friday, June 6, a Process Improvement Track is offered featuring several of CSC's process improvement experts. Saturday, June 7, the "Process Improvement Across CSC" exhibit is available for you to speak directly to your CSC process improvement experts and learn more about what methodologies/models will best benefit your project/organization needs.
Visit the CSC Global Technology & Business Solutions Conference Community for schedule and registration details. Sign up today!
News 1
Profile
Community Purpose
The purpose of the CSC Global Process Improvement Community will be to serve as a "parent" community to all content-specific process improvement communities. Anyone interested in investigating the various process
Internet
Start
52°
10:20 AM

Step 6 – Execute Your Processes



Strategy: Base Your Business Activities on Your Defined Processes

- Institutionalize your processes
  - Train your workforce in your defined processes
  - Incentivize both management and staff
- Check process performance
  - Goal ⇒ Process ⇒ Results (GPR)
    - A business-focused Plan ⇒ Do ⇒ Check ⇒ Act cycle
    - For a given set of results either goals or processes may be modified



Technology: Use Tools to Facilitate Process Deployment, Training, and Execution

- Process Asset Libraries
- Knowledge Portals
- Computer-Based Training (CBTs)
- Web-Based Training (WBT)
- Net-Based Collaboration
  - MS Windows NetMeeting
    - http://www.microsoft.com/windows/netmeeting/default.asp
  - AT&T Web Meeting
    - http://www.business.att.com/products/productdetails.jsp;jsessionid=PM0HJK2YN2ZQFLAZBYZCFEY?productId=wms_eu
  - IBM Lotus Sametime
    - http://www.lotus.com/products/lotussametime.nsf/wdocs/homepage

Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

Step 7 – Measure Your Results - Modify Processes as Necessary

Strategy: Establish an Active Measurement Program

- For both Processes and Products:
  - **Identify** measurement objectives that are aligned with your business needs
  - **Choose** specific measures, data collection and storage mechanisms, analysis techniques, and reporting and feedback mechanisms that support your measurement objectives
  - **Implement** your measurement program
  - **Provide** decision-makers with objective results that can be used in making informed decisions
  - **Periodically assess** your measurement program to ensure that it is meeting current business needs

Technology: Use Tools to Support Measurement of Process Performance and Product Quality

- Spreadsheets
- Control Charts
  - http://www.isixsigma.com/st/control_charts/
- PSM Insight
  - http://www.psmsc.com/PSMI.asp
- Distributive Software's DataDrill
  - http://www.distributive.com/solutions_maturitymodel.html
- SEI Software Engineering Measurement and Analysis Program
  - http://www.sei.cmu.edu/sema/welcome.html
- INCOSE Measurement Working Group
  - http://www.incosemwg.org/

Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

Step 8 – Confirm Your Status With Independent Appraisals

Strategy: Select an Appraisal Style Suited to Your Process Improvement Objectives

Attributes	***Class A***	***Class B***	***Class C***
Usage Mode	**• Benchmark • Baseline establishment**	**• Initial • Incremental • Self-assessment**	**• Quick Look • Incremental • Gap analysis**
Relative: • Cost/Duration • Confidence • Accuracy	**High**	**Medium**	**Low**
Rating?	**Yes**	**No**	**No**

http://www.sei.cmu.edu/cmmi/appraisals/appraisals.html

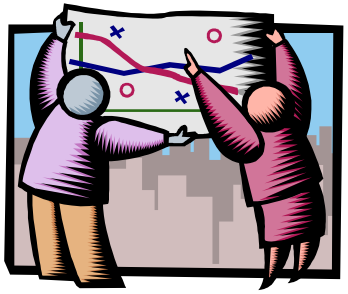
Technology: Use Process Appraisal Tools to Capture Appraisal Data and Facilitate Findings

- Checklists
- Spreadsheets
- Databases
- Integrated Systems Diagnostics' Appraisal Wizard
  - http://www.isd-inc.com/index.asp?q=1036&
- pragma Systems Corporation's processMax[5]
  - http://www.pragmasystems.com/max5fin.htm
- SEI Transition Partner Appraisal Services
  - http://www.sei.cmu.edu/collaborating/partners/partners-tech.html#SCAMPI

Disclaimer: Tools and services mentioned are for example only. Such mention is not to be construed as an endorsement by the author, CSC, IEEE, ISO. or IEC

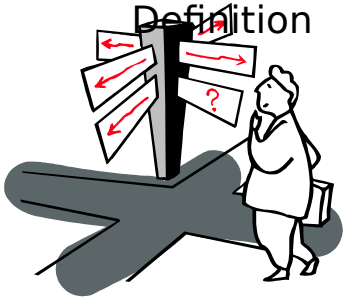
8 Steps to Success In CMMI®-Compliant Process Engineering

1. Understand your business processes



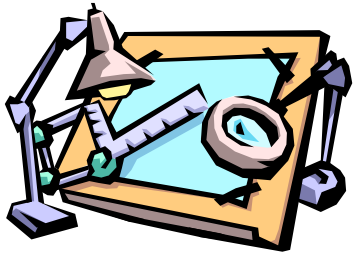
2. Look to the CMMI℠ for Process Completenes

3. Look to Framework Standards for Life Cycle Definition

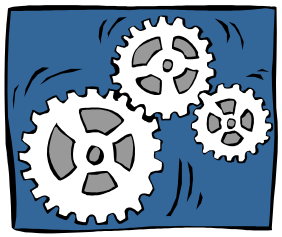


4. Look to Supporting Standards for Process Detail

5. Build or Refine Your Process Architecture



6. Execute Your Processes



7. Measure Your Results - Modify Processes as Necessary

8. Confirm Your Status With Independent Appraisals

For More Information . . .

Paul R. Croll
Computer Sciences Corporation
5166 Potomac Drive
King George, VA 22485-5824

Phone: +1 540.644.6224
Fax: +1 540.663.0276
e-mail: pcroll@csc.com

For IEEE Standards:

http://computer.org/standards/sesc/

http://computer.org/cspress/CATALOG/st01110.htm

For ISO/IEC Standards:

http://saturne.info.uqam.ca/Labo_Recherche/Lrgl/sc7/

References

CMMI® -SE/SW/IPPD/SS, V1.1, *CMMI for Systems Engineering/Software Engineering/Integrated Product and Process Development, and Supplier Sourcing Version 1.1, CMMI℠ -SE/SW/IPPD/SS, V1.1, Continuous Representation.* CMU/SEI-CMU/SEI-2002-TR-011, ESC-TR-2002-011, Carnegie Mellon University, Software Engineering Institute, Pittsburgh, PA, March 2002.

IEEE Standard 830-1998, *Recommended Practice for Software Requirements Specifications,* Institute of Electrical and Electronics Engineers, Inc. New York, NY, 1998.

IEEE Standard 1233-1998, *Guide for Developing System Requirements Specifications,* Institute of Electrical and Electronics Engineers, Inc. New York, NY, 1998.

References - 2

IEEE/EIA Standard 12207.0-1996, *Industry Implementation of International Standard ISO/IEC12207:1995 — (ISO/IEC 12207) Standard for Information Technology —Software life cycle processes,* Institute of Electrical and Electronics Engineers, Inc. New York, NY, 1998.

IEEE/EIA Standard 12207.1-1997, *Industry Implementation of International Standard ISO/IEC12207:1995 — (ISO/IEC 12207) Standard for Information Technology —Software life cycle processes – Life cycle data,* Institute of Electrical and Electronics Engineers, Inc. New York, NY, 1998.

IEEE/EIA Standard 12207.2-1997, *Industry Implementation of International Standard ISO/IEC12207:1995 — (ISO/IEC 12207) Standard for Information Technology — Software life cycle processes – Implementation considerations,* Institute of Electrical and Electronics Engineers, Inc. New York, NY, 1998.

References - 3

ISO/IEC 15288:2002, *Systems Engineering — System Life Cycle Processes*, ISO/IEC JTC1/SC7, 2002.

ISO/IEC 19760:2003, *Systems Engineering – A Guide for the application of ISO/IEC 15288 System Life Cycle Processes,* ISO/IEC JTC1/SC7, 2003.

[Singh97] Raghu Singh, *An Introduction to International Standards ISO/IEC 12207, Software Life Cycle Processes*, 1997.